MASTERFUL

Monotonic Adaptive Solutions of Transient Equations using Recovery, Fitting, Upwinding and Limiters

M.J. Baines

Numerical Analysis Report 5/92

Department of Mathematics
University of Reading
P.O. Box 220
Reading, UK

The work reported here forms part of the research programme of the Oxford/Reading Institute for Computational Fluid Dynamics.

Abstract

A method of approximately solving transient, particularly convective, equations on adaptive grids is presented, which has the useful properties (i) that both the grid and the solution remain monotonic, and (ii) that the solution is always the best fit with adjustable nodes to a recovered smoother version of itself. In this report the underlying representation is piecewise constant and the recovered function piecewise linear, while the equation is the inviscid Burgers' equation solved by upwind finite differences, but other representations, equations and schemes are also discussed.

Contents

1 Introduction

The solutions of time-dependent partial differential equations may exhibit steep fronts whose locations vary with time and are difficult to predict in advance. Resolution of the resulting profiles can be achieved by a fine grid everywhere, but such a grid is impractically large, especially for 3-D calculations.

An adaptive grid is therefore demanded, one which will ideally follow the fronts around and provide the required resolution in a selective way. In order to achieve this ability, however, two difficult questions must be addressed. The first is the problem of representation of the solution on an irregular grid or rather, in this context, the problem of knowing the best irregular grid to represent the solution. The second is the control of the grid as the solution changes position and shape, in particular the prevention of node overtaking (which is the equivalent in nodal terms of monotonicity in the solution profile).

The first problem has been tackled by a number of authors in the context of approximation theory, in particular de Boor [1], Chui [2], Loach & Wathen [3], Farmer, Heath & Moody [4] and Baines [5]. The latter gives algorithms for obtaining best L_2 fits to continuous functions using piecewise constant and piecewise linear representations with adjustable nodes which are relatively simple and robust and are the basis of the nodal movement used here. Of course in the present context there is no given function to be fitted, since the exact solution of the PDE is not known, but we get round this difficulty by predicting the new numerical solution and makinguse of a recovered function (see below).

The second problem, that of nodal movement and its control, has also been the subject of many papers, in particular Dorfi & Drury [6], Petzold [7] and Miller [8] (see also Verwer et al. [9]). The moving finite element (MFE) method method of Miller [8] is an attempt to move the nodes by the same mechanism which controls the solution, namely, consistency with the underlying PDE. As shown by Baines [10], the result for first order PDE's is a characteristic0following method (akin to a Hamiltonian approach), but in the case of higher order PDE's the nodal velocities generated are less well understood and their effectiveness is more dubious, except in the steady state limit (Jimack [13]). In any case Miller [8] modifies nodal movement by using penalty functions to control node overtaking and other singularities which may appear. This has the effect of maintaining monotonicity of the grid as required, but destroys the approximate characteristic nodal speeds property, under which the nodes are expected to overtake [11]. It does however assist the effect of nodal speeds arising from higher order terms in restrianing node overtaking.

The MFE method has been analysed further by baines [12], bringing out a link between the equations used to generate nodal movement and those giving best fits with adjustable nodes (see also [4], [5] and [13]). From the analysis in [12] it is clear that MFE nodal velocities for first order equations are made of up to two parts. First there is the approximation to the characteristic velocity, apparently already in the inviscid Burgers' equation where the L_2 projection step of MFE is superfluous [14], [10]. Secondly, there is an additionla velocity generated by the projection step which has been equated with one step of the iterative algorithm to find the best fit with adjustable nodes described in [5].

The characteristic velocity draws the nodes into shocks where they naturally coalesce, or overtake if not prevented by diffusive effects, real or artificial. (Even with real viscosity Miller [8] requires penalty functions with tuning parameters to prevent node overtaking, a device which tends to be problem dependent and reduce the method's usefulness.) On the other hand, the best-fit-seeking velocity coming from the L_2 projection maintains monotonicity in the grid [5] while at the same time clustering nodes in the regions of high slopes or curvature (depending on whether the underlying representation is piecewise constant or piecewise linear).

In this report we construct a method whose grid velocity is derived from the L_2 projection alone. Care is needed, however, because where no projection is needed, as for example with the inviscid Burgers' equation, there will be no grid velocity. This objection is countered by projecting instead a smoother "recovered" function (Johnson et al. [15]) which is in the form of a locally constructed polynomial.

When imposing a grid velocity on the nodes the differential equation must be modified to take into account the fact that it is now set in Lagrangian (moving) frame, the PDE being augmented by an extra convective term. Moreover, in solving PDE's with convective terms it is usually necessary to employ some kind of upwind differencing to achieve monotonicity. By going for monotonicity (both in the solution and the grid) the numerical viscosity generated provides the usual non-overturned shock-like approximation to the solution of a convective equation and its corresponding jump condition, but will be less than for a fixed grid.

In the next section we concentrate on a particular discrete solution representation and work through the method proposed above for a particular equation, later indicating the possible generalisations and their properties.

2 Piecewise Constants in 1-D

Suppose that the underlying discrete representation of the solution of a PDE is piecewise constant, as shown in Fig. 1.

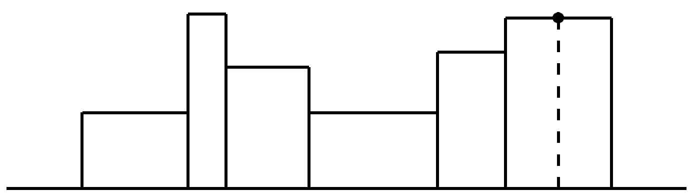

Figure 1:

Individual point values are thought of as existing at the midpoints of each element as in the rightmost cell of Fig. 1. We consider here the Cauchy problem for the inviscid Burgers' equation problem

$$u_t + uu_x = 0 \quad (t > 0) \tag{2.1}$$

$$u = u_0(x) \quad (t = 0) \tag{2.2}$$

A non-adaptive scheme for this problem is the upwind finite difference method

$$U_j^{n+1} = U_j^n - |U_j^n| \frac{\Delta t}{\Delta_o X_j^n} (U_j^n - U_{j-\mu}^n) \tag{2.3}$$

where $\mu = \text{sgn}(U_j^n)$ and $\Delta_o X_j^n = \frac{1}{2}(X_{j+1}^n - X_{j-1}^n)$. (The right hand side of (2.3) represents a linear interpolation of the values of U_j^n after being traced back along the characteristics.) While consistent with the differential equation (2.1), the scheme (2.3) is highly diffusive and, particularly in the case of a steepening wave, gives very poor representation of the exact solution. The philosophy of adaptive grids is that sharper shocks can be achieved by clustering of the grid points in a controlled way, namely, in the vicinity of the shock, and for this we use the grid movement strategy described earlier.

The first task, however, is to represent the function $u_0(x)$ of (2.2) as well as possible using piecewise constant functions on a grid with free nodes. This problem is addressed by Baines [5], where a simple algorithm is given to generate such a representation. In summary, this is as follows. For a given continuous function $f(x)$, an initial arbitrary grid is set up in each element of which a constant approximation is found by a local L_2 projection (Fig. 2).

Figure 2:

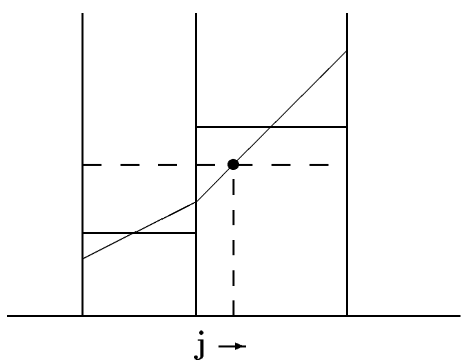

Figure 3:

Then, taking two adjacent elements, the averaged constant value is calculated (shown dotted in Fig. 2) and its intersection with the $f(x)$ function found. All grid points are treated in this way in a sweep, before another sweep is done. At convergence the best representation is obtained. A key property of the construction is that the nodes remain ordered, i.e. there is no node tangling.

The procedure also achieves the same result when $f(x)$ is replaced by the piecewise linear interpolant function, with values $f(X_j)$ at X_j as in Fig. 3. This is important for what follows.

Having generated a best initial grid and profile, the next task is to try and maintain this property as the solution evolves. As is well known, an application of the algorithm (2.3) on a fixed grid results in a profile of the correct general shape, but diffused. We wish to use this (piecewise constant) profile to determine a suitable grid movement. The best fit algorithm of Baines [5] operates on a continuous function (or a piecewise linear continuous function), which we do not have. But we can construct a piecewise linear recovered function from the new piecewise constant function, which is shown in Fig. 4 as the piecewise linear function linking the mid-points of the piecewise constant cell values.

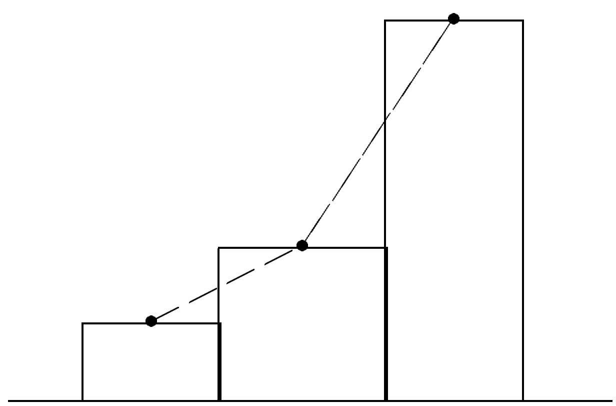

Figure 4:

As explained above, the algorithm for finding the best piecewise constant fit to this piecewise linear function is in the form of an iteration, the first step of which gives a new grid position X_j, which may be regarded as being carried out by a certain grid movement $\dot{X}_j$. The formula for the new grid position X_j is easily obtained, being the intersection between the piecewise linear function and the ordinate

$$\frac{1}{2}\left[\frac{1}{2}(U^n_{j+1}+U^n_j)+\frac{1}{2}(U^n_j+U^n_{j-1})\right]=U^n_j+\frac{1}{4}\delta^2 U_j. \tag{2.4}$$

The new position is therefore

$$X^{n+1}_j=\begin{cases} X^n_j+\frac{1}{4}\frac{\delta^2 U^n_j}{M^n_{j-\frac{1}{2}}} & |U^n_j-U^n_{j-1}|\geq|U^n_{j+1}-U^n_j| \\ X^n_j+\frac{1}{4}\frac{\delta^2 U^n_j}{M^n_{j+\frac{1}{2}}} & |U^n_j-U^n_{j-1}|<|U^n_{j+1}-U^n_j| \end{cases} \tag{2.5}$$

where

$$M^n_{j-\frac{1}{2}}=\frac{U^n_j-U^n_{j-1}}{X^n_j-X^n_{j-1}},\quad M^n_{j+\frac{1}{2}}=\frac{U^n_{j+1}-U^n_j}{X^n_{j+1}-X^n_j} \tag{2.6}$$

Where two intersections are possible the node is moved the shorter distance in the direction of the higher slope.

We do not carry out any iteration, since corrections will depend strongly on the linear nature of the recovery. The scheme, iteration and recovery should be of comparable accuracy. We now return to the PDE, rewritten in a moving frame as

$$\dot{u}-u_x\dot{x}+uu_x=0 \tag{2.7}$$

(Baines [12], Blom et al. [9]), where $\dot{x}$ is the velocity of the frame and $\dot{u}$ is the time derivative in that frame, and use it to generate new values U^n_j. We do this by the algorithm (2.3) modified to incorporate the extra velocity, namely,

$$U^{n+1}_j=U^n_j-\left|U^n_j-\dot{X}^{(1)}_j\right|\frac{\Delta t}{\Delta_o X^n_j}(U^n_j-U^n_{j-\mu}) \tag{2.8}$$

where $\mu=\mathrm{sgn}(U^n_j-\dot{X}^{(1)}_j)$. The velocity $\dot{X}^{(1)}_j$ is given by

$$\dot{X}^{(1)}_j=\frac{X^{(1)}_j-X_j}{\Delta t} \tag{2.9}$$

where $X^{(1)}_j-X_j$ is the displacement generated by the best fit algorithm (2.5) above. Note that the effect of the $u_x\dot{x}$ term is simply to modify the wavespeed u.

Thus we have

$$X^{n+1}_j=\begin{cases} X^n_j+\frac{1}{4}\frac{\delta^2 U_j}{M^n_{j-\frac{1}{2}}} & |U^n_j-U^n_{j-1}|\geq|U^n_{j+1}-U^n_j| \\ X^n_j+\frac{1}{4}\frac{\delta^2 U_j}{M^n_{j+\frac{1}{2}}} & |U^n_j-U^n_{j-1}|<|U^n_{j+1}-U^n_j| \end{cases} \tag{2.10}$$

$$U^{n+1}_j=U^n_j-\left|U^n_j-\frac{(X^{n+1}_j-X^n_j)}{\Delta t}\right|\frac{\Delta t}{\Delta_o X^n_j}(U^n_j-U^n_{j-\mu}) \tag{2.11}$$

where

$$\mu = \left\{ U_j^n - \frac{(X_j^{n+1} - X_j^n)}{\Delta t} \right\}. \tag{2.12}$$

The result is a piecewise constant profile at time level $n+1$, obtained by a finite difference scheme consistent with the PDE, which is a best L_2 fit to the piecewise linear recovered function (standing in lieu of the exact solution) at time level $n+1$.

3 Monotonicity of the Algorithm

The algorithm of section 2 is

$$X_j^{n+1} = \begin{cases} X_j^n + \frac{1}{4}\frac{\delta^2 U_j^n}{M_{j-\frac{1}{2}}^n} & |U_j^n - U_{j-1}^n| \geq |U_{j+1}^n - U_j^n| \\ X_j^n + \frac{1}{4}\frac{\delta^2 U_j^n}{M_{j+\frac{1}{2}}^n} & |U_j^n - U_{j-1}^n| < |U_{j+1}^n - U_j^n| \end{cases} \tag{3.1}$$

$$U_j^{n+1} = U_j^n - \left| U_j^n - \frac{(X_j^{n+1} - X_j^n)}{\Delta t} \right| \frac{\Delta t}{\Delta_o X_j^n}(U_j^n - U_{j-\mu}^n) \tag{3.2}$$

where

$$\left. \begin{array}{c} \mu = \operatorname{sgn}\left\{ U_j^n - \frac{(X_j^{n+1} - X_j^n)}{\Delta t} \right\}, \ M_{j-\frac{1}{2}}^n = \frac{U_j^n - U_{j-1}^n}{X_j^n - X_{j-1}^n} \\ \\ \Delta_o X_j^n = \frac{1}{2}(X_{j+1}^n - X_{j-1}^n) \end{array} \right\} \tag{3.3}$$

The geometrical construction of X_j^{n+1} ensures that

$$X_j^n \leq X_j^{n+1} \leq \frac{1}{2}(X_j^n + X_{j+1}^n) \tag{3.4}$$

for

$$M_{j-\frac{1}{2}}^n \leq M_{j+\frac{1}{2}}^n, \ \ |U_{j+1}^n - U_j^n| \geq |U_j^n - U_{j-1}^n|$$

and

$$\frac{1}{2}(X_{j-1}^n + X_j^n) \leq X_j^{n+1} \leq X_j^n \tag{3.5}$$

for

$$M_{j-\frac{1}{2}}^n \geq M_{j+\frac{1}{2}}^n, \ \ |U_j^n - U_{j-1}^n| < |U_{j+1}^n - U_j^n|$$

These inequalities can also be proved analytically from (3.1). No tangling is therefore possible and monotonicity of the mesh is achieved. Moreover the upwinded nature of the scheme (3.2) means that, provided that a CFL limit is respected, monotonicity of u_j is also preserved. The CFL restriction is that

$$\left| U_j^n - \frac{(X_j^{n+1} - X_j^n)}{\Delta t} \right| \frac{\Delta t}{\Delta_o X_j^n} \leq 1 \quad \forall j \tag{3.6}$$

The mesh movement is always towards the steeper part of the profile and, with monotonicity in both the solution and the grid, the profile is literally forced into a corner. We illustrate this phenomenon on equation (2.1) with the initial data

$$u_0(x) = -x + \frac{1}{2} \tag{3.7}$$

on (0,1), with Dirichlet boundary conditions taken from the initial data. Figures 5-12 show the initial data and the solution after 5, 10, 15, 20, 25, 30, and 35 time steps.

4 Conservation of the Algorithm and Higher Order

In the scheme (2.3), (2.8) and (2.11) we have concentrated on exhibiting monotonicity preservation and ignored conservation. But we may ensure conservation instead by replacing (2.3) by the conserved form of the same scheme, namely,

$$U_j^{n+1} = U_j^n - \mu U_{j-\frac{\mu}{2}}^n \frac{\Delta t}{\Delta_o X_j^n} (U_j^n - U_{j-\mu}^n) \tag{4.1}$$

where

$$U_{j-\frac{\mu}{2}}^n = \frac{1}{2}(U_j^n + U_{j-\mu}^n), \tag{4.2}$$

$$\mu = \operatorname{sgn}(U_{j-\frac{\mu}{2}}^n) \tag{4.3}$$

or, equivalently,

$$U_j^{n+1} = U_j^n - \mu \frac{\Delta t}{\Delta_o X_j^n} \left\{ \frac{1}{2}(U_j^n)^2 - \frac{1}{2}(U_{j-\mu}^n)^2 \right\} \tag{4.4}$$

where terms are included for all $\mu = \pm 1$ (usually one or the other) which satisfy (4.3).

Similarly, (2.8) becomes

$$U_j^{n+1} = U_j^n - \mu \frac{\Delta t}{\Delta_o X_j^n} \left\{ \frac{1}{2}(U_j^n)^2 - \frac{1}{2}(U_{j-\mu}^n)^2 \right\}$$

$$+ \mu \frac{\Delta t}{\Delta_o X_j^n} \left\{ \dot{X}_j U_j^n - \dot{X}_{j-\mu} U_{j-\mu}^n \right\} \tag{4.5}$$

where now

$$\mu = \operatorname{sgn}(U_{j-\frac{\mu}{2}}^n - \dot{X}_j). \tag{4.6}$$

In the final bracket of (4.5) the term $\dot{x}u_x$ in the differential equation has been written as $(\dot{x}u)_x$ to render the equation (4.5) conservative. There is no loss of generality here since the additional term incurred, even if non-zero in its numerical form, disappears in the analytic limit (Hyman [16]). In the numerical scheme the effect of the term $(\dot{x}u)_x$ in the PDE is simply to modify the wavespeed.

The form of (2.11) is then

$$U_j^{n+1} = U_j^n - \mu \frac{\Delta t}{\Delta_o X_j^n} \left\{ \frac{1}{2}(U_j^n)^2 - \frac{1}{2}(U_{j-\mu}^n)^2 \right\}$$

$$+ \mu \frac{\Delta t}{\Delta_o X_j^n} \left\{ (X_j^{n+1} - X_j^n) U_j^n - (X_{j-\mu}^{n+1} - X_{j-\mu}^n) U_{j-\mu}^n \right\} \tag{4.7}$$

with

$$\mu = \operatorname{sgn} \left\{ [U_{j-\frac{\mu}{2}}^n - (X_{j-\frac{\mu}{2}}^{n+1} - X_{j-\frac{\mu}{2}}^n)]/\Delta t \right\} \tag{4.8}$$

where

$$X_{j-\frac{\mu}{2}} = \frac{1}{2}(X_j + X_{j-\mu}). \tag{4.9}$$

Monotonicity depends on satisfying a CFL condition, which is now

$$\left| U_{j-\frac{\mu}{2}}^n - (X_{j-\frac{\mu}{2}}^{n+1} - X_{j-\frac{\mu}{2}}^n)/\Delta t \right| \frac{\Delta t}{\Delta_o X_j^n} \leq 1 \quad \forall j \tag{4.10}$$

for any $\mu = \pm 1$ satisfying (4.8).

Second order accuracy on a uniform grid may be achieved using flux limiters (Smolarkiewicz & Margolin [17]). However, the irregularity of the grid and indeed its capacity to adapt in the present manner may replace the need for this refinement near to a shock.

5 Relationship With Modified Equation

For the first order upwind scheme (2.3) applied to

$$u_t + a u_x = 0 \tag{5.1}$$

with $a > 0$, Taylor expansion gives

$$u_t + au_x = \frac{1}{2}a\frac{(\Delta^- X)^2}{\Delta_0 X}(1-\nu)u_{xx} + \frac{\delta^2 X}{\Delta_0 X}au_x + O((\Delta x)^2, (\Delta t)^2) \tag{5.2}$$

where $\Delta^- X = X_j - X_{j-1}$, $\Delta_0 X = \frac{1}{2}(X_{j+1} - X_{j-1})$, $\delta^2 X = X_{j+1} - 2X_j + X_{j-1}$, and

$$\nu^- = \frac{a\Delta t}{\{(\Delta^- X)^2/\Delta_0 X\}}$$

In a Langragian frame the extra terms $\dot{x}u_x$ match the additional term in (5.2) if

$$\begin{aligned}\dot{x} &= \frac{1}{2}a\frac{(\Delta^- X)^2}{\Delta_0 X}(1-\nu^-)\frac{u_{xx}}{u_x} + \frac{a\delta^2 X}{\Delta_0 X} + O((\Delta x)^2, (\Delta t)^2) \\ &= \frac{1}{2}a\frac{(\Delta^- X)^3}{\Delta_0 X}(1-\nu^-)\frac{\delta^2 U}{\Delta^- U} + \frac{a\delta^2 X}{\Delta_0 X}\end{aligned} \tag{5.3}$$

where $\Delta^- U = U_j - U_{j-1}$, or to the same order,

$$X_j^{n+1} - X_j^n = \frac{1}{2}\left[\frac{\Delta^- X}{\Delta_0 X}\right]^2 \nu^-(1-\nu^-)\frac{\delta^2 U_j}{M_{j-\frac{1}{2}}} + \left[\frac{\Delta^- X}{\Delta_0 X}\right]^2 \nu^- \delta^2 X \tag{5.4}$$

c.f. (2.5). This is related to the idea of Smolarkiewicz and Margolin [17].

Similarly, for $a > 0$, to the same order,

$$X_j^{n+1} - X_j^n = \frac{1}{2}\left[\frac{\Delta^+ X}{\Delta_0 X}\right]^2 \nu^+(1-\nu^+)\frac{\delta^2 U_j}{M_{j+\frac{1}{2}}} + \left[\frac{\Delta^+ X}{\Delta_0 X}\right]^2 \nu^+ \delta^2 X \tag{5.5}$$

where $\nu^+ = a\Delta t/(X_j^{n+1} - X_j^n)$.

Comparing (5.4), (5.5) with (2.5), we see that the displacement of (2.5) also represents a diffusive correction. Moreover, the terms $\frac{1}{2}\nu^{\pm}(1-\nu^{\pm})$ in (5.4) or (5.5) lie betwen 0 and $\frac{1}{8}$ (with $|\nu^{\pm}| < 1$) and non-overtaking property inherent in (2.5) is enforced in the appropriate terms of (5.4), (5.5) a *fortiori.*

6 Alternative Representation

In section 2 we chose piecewise constant representation of the initial data but this is not the only possibility. A piecewise linear representation (or higher order) can be used. All that is required is the availability of a best fitting algorithm with adjustable nodes for a recovered function, as can be found in Baines [5] for piecewise linears, and Loach [18] for higher order splines, and a monotonicity-preserving algorithm (Leonard et al. [19], Priestly [20]). For example, recovery of piecewise linears can be achieved using limited Hermite cubics as in Priestly [21] and all the ingredients of the method are then in place. The piecewise linear best fit algorithm of Baines [5] again involves no tangling of the grid, but this time the movement of the grid is towards greater curvature in the profile rather than greater steepness (Baines [5]).

While the method presented here is directed mainly at convective equations where node clustering is required near fronts, there is no reason why other equations cannot be treated by the same method.

References

[1] de Boor, C. (1973). Good approximation by splines with variable knots, In Spline Functions and Approximation Theory, Int. Ser. Num. Meths., **21**, Basel, Birkhauser.

[2] Chui, C.K., Smith, P.W. & Ward, J.D. (1977). On the Smoothness of Best L_2 Approximants from Nonlinear Spline Manifolds, Math. Comp., **31**, 17-23.

[3] Loach, P.D. & Wathen, A.J. (1991). On the Best Least Squares Approximation of Continuous Functions using Linear Splines with Free Knots, IMA Journal of Num. An., **11**, 393-409.

[4] Farmer, C.L., Heath, D. and Moody, R.O. (1991). A Global Optimisation Approach to Grid Generation, SPE 21236. In Proc. of 11th SPE symposium on Reservoir Simulation, Anaheim.

[5] Baines, M.J. (1991). Algorithms for Best Piewise Discontinuous Linear and Constant L_2 Fits to Continuous Functions with Adjustable Nodes in One and Two Dimensions. Submitted to SIAM J. of Scientific and Statistical Computing.

[6] Dorfi, E.A. & Drury, LO'C. (1987). Simple Adaptive Grids for 1-D Initial Value Problems. J. Comput. Phys. **69**, p 175.

[7] Petzold, L. (1987). Observations on an Adaptive Moving Grid Method for 1-D systems of PDE's, Appl. Num. Math., **3** p 347.

[8] Miller, K., (1981). Moving Finite Elements I (with R.N. Miller), Moving Finite Elements II, SIAM J. Num. Anal. **18**, p1019-1957.

[9] Blom, J.G., Sanz-Serna, J.M. and Verwer, J.G., (1988). On Simple Moving Gris Methods for One Dimensional Evolutionary PDE's, J. Comput. Phys. **46**, p 342-368.

[10] Baines, M.J. (1991). An Analysis of the Moving Finite Element Procedure, SIAM J. Num. Anal. **28**, p1323-1349.

[11] Miller, K. (1986). Alternative Modes to Control the Nodes in the Moving Finite Element Method. In Adaptive Computational Methods for PDE's (Ed. Babuska, Chandra & Flaherty). SIAM p 165-182.

[12] Baines, M.J. (1992). On the Relationship between the Moving Finite Element Method and the Best Fits to Functions with Adjustable Nodes, Numerical Analysis Report 2/91, Dept. of Mathematics, University of Reading, U.K.

[13] Jimack, P. (1992). Large Time Solutions of the Moving Finite Element Equations for Linear Diffusion Problems in 1-D, IMA J. Num. Anal. (to appear).

[14] Wathen, A.J. & Baines, M.J. (1985). On the Structure of the Moving Finite Element Equations. IMA J. Num. Anal. **5**, p 161-182.

[15] Johnson, I.W., Wathen, A.J. & Baines, M.J. (1988). Moving Finite Element Methods for Evolutionary Problems II, Applications. J. Comput. Phys. **79**, p 270.

[16] Hyman, J.M., (1989). Private Communication, SIAM Conference, San Diego.

[17] Smolarkiewicz, P. & Margolin, L., (1989). Antidiffusive Velocities for Multi-Pass Donor Cell Advection. Report No. UCID 21866. Lawrence Livermore National Laboratory.

[18] Loach, P., (1992). Private Communication, School of Mathematics, University of Bristol.

[19] Leonard, B.P., et al. (1991). Sharp Monotonic Resolution of Discontinuities Without Clipping of Narrow Extrema. Computers and Fluids **19**, p 141-154.

[20] Priestly, A. (1990). A Quasi-Riemann Method for the Solution of One-Dimensional Shallow Water Flow. Numerical Analysis Report 5/90, Dept. of Mathematics, University of Reading, U.K.

[21] Priestly, A. (1991). PLAGIARISM - The Monotonic and Nearly Conservative Lagrange-Galerkin Method. Numerical Analysis Report 15/91, Dept. of Mathematics, University of Reading, U.K.